REMARKS

This is intended as a full and complete response to the Office Action dated January 11, 2005, having a shortened statutory period for response set to expire on April 11, 2005. Reconsideration and allowance of the claims pending in the application is requested for reasons discussed below.

In this Office Action, claims 5-13 were provisionally rejected for obviousness type double patenting as unpatentable over claims 1 and 36-41 of co-pending application 10/609,967. As the Examiner notes, neither the claims in this application nor the co-pending application are allowed. Therefore, consideration of this rejection is requested to be held in abeyance until the prosecution is successfully concluded.

Claims 1-13 are rejected under 35 U.S.C. 103(a) as obvious over U.S. Patent 6,630,935 considered with *Rentschler*, U.S. Patent 5,969,726. These rejections are respectfully traversed. It is respectfully submitted that the references cited by the Examiner fail to teach the present invention as claimed.

In this response, the claims originally presented have been edited to clarify the present invention, and additional claims 14-20 are presented herein to provide the inventor the full scope of protection to which he is entitled.

As originally presented, and as now edited for clarity, the present claims cover an invention including a multithreaded processor for processing different sample types, controlled by different instruction sets, in a plurality of threads. More specifically, a multithreaded processor is described which is configured to execute programs associated with a multiplicity of threads. Each thread is capable of simultaneously executing operations on any one of multiple sample types which in graphics processing may comprise pixels, fragments, vertex, or other data. Such independent processing optimizes the speed of execution of the processing, depending on what type of sample data may be received.

The processor provides for allocating the number of threads assigned to each sample of a certain type on a cycle-to-cycle basis; such allocation, which may be fixed, dynamic, programmable or the like, permits load balancing to modify the ratio of processing units assigned to pixel data, vertex data or the like. Further, the use of such

programmability of the independent threads allows for interleaving the execution of programs associated with samples of different types in the multiple threads within each processing unit. The type of thread being executed can vary from one clock cycle to the next. Thus, each thread can successively execute instructions of different types associated with samples of different types of data; and the multithreads can simultaneously execute different instruction sets associated with different data sample types depending on the amount of data of different sample types being received. The allocation of any one sample is limited by a global state value which may be fixed or programmable.

The references cited by the Examiner, considered individually or in combination, do not teach any or all of the features summarized above. *Taylor* teaches only the use of threads of data which are fed to an arbitration module 14. The arbitration module responds to an application-specific prioritization scheme to feed the data received from the thread controllers 18-24 and now associated with an operation code 48 to a computation processor 12 to optimize the use of the computation engine. While *Taylor* teaches multiple threads, the reference does not teach simultaneous processing of multiple threads, the data samples being constantly and dynamically allocated among multiple processors for simultaneous execution, with the allocation being modified to optimize the efficiency of processing. *Taylor* further does not teach or suggest that the thread control unit may apply different instruction sets successively to the same thread, or that the same thread may successively process data samples of different types in different clock cycles. Moreover, the Examiner does not allege that such a teaching can be found therein.

As to the citation of *Rentschler*, although *Rentschler* does disclose a "global state value" as being a value maintained and utilized for processing all vertices in an image being rendered, he does not teach the specific type of global state value disclosed and claimed herein. As described, for example, at paragraphs 60 and 61 of the present application, the sample portion global state value is a specific type of global state value which is utilized to cap the amount of data which can be subject to allocation in a specific embodiment of the multithread processor. *Rentschler* does not teach

allocation, or any of the principles used in allocation of sample types. Therefore, *Rentschler* adds nothing to the combination cited and relied on by the Examiner in terms of teaching the claimed invention.

In view of the clear distinctions between the claims as now entered and the art cited, reconsideration and allowance of the claims is requested.

Having addressed all issues set out in the office action, Applicants respectfully submit that the claims are in condition for allowance and respectfully request that the claims be allowed.

Respectfully submitted,

James A. Sheridan

James A. Sheridan, Reg. No. 25,435
MOSER, PATTERSON & SHERIDAN, L.L.P.
3040 Post Oak Blvd. Suite 1500
Houston, TX 77056
Telephone: (713) 623-4844
Facsimile: (713) 623-4846
Attorney for Applicant(s)